AF JFW

TRANSMITTAL OF APPEAL BRIEF

Docket No. BBNT-P01-128

In re Application of: Milliken et al.

Application No.	Filing Date	Examiner	Group Art Unit
09/938921	August 24, 2001	Q. N. Nguyen	2141

Invention: TERNARY CONTENT ADDRESSABLE MEMORY EMBEDDED IN A CENTRAL PROCESSING UNIT

TO THE COMMISSIONER OF PATENTS:

Transmitted herewith is the Appeal Brief in this application, with respect to the Notice of Appeal filed: June 21, 2005 .

The fee for filing this Appeal Brief is $ 500.00 .

[x] Large Entity [ ] Small Entity

[ ] A petition for extension of time is also enclosed.

The fee for the extension of time is _____ .

[ ] A check in the amount of _____ is enclosed.

[x] Charge the amount of the fee to Deposit Account No. 18-1945 .
This sheet is submitted in duplicate.

[ ] Payment by credit card. Form PTO-2038 is attached.

[x] The Director is hereby authorized to charge any additional fees that may be required or credit any overpayment to Deposit Account No. 18-1945 .
This sheet is submitted in duplicate.

Dated: August 2, 2005

Edward A. Gordon
Attorney Reg. No.: 54,130
ROPES & GRAY LLP
One International Place
Boston, Massachusetts 02110-2624
(617) 951-7066

I hereby certify that this correspondence is being deposited with the U.S. Postal Service with sufficient postage as First Class Mail, in an envelope addressed to: MS Appeal Brief - Patents, Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450, on the date shown below.

Dated: 8/2/05 Signature: Joanne Ryan (Joanne Ryan)

I hereby certify that this correspondence is being deposited with the U.S. Postal Service with sufficient postage as First Class Mail, in an envelope addressed to: MS Appeal Brief - Patents, Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450, on the date shown below.

Dated: 8/2/05 Signature: Joanne Ryan (Joanne Ryan)

Docket No.: BBNT-P01-128
(PATENT)

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Patent Application of:
Milliken et al.

Application No.: 09/938921 Confirmation No.: 3501

Filed: August 24, 2001 Art Unit: 2141

For: TERNARY CONTENT ADDRESSABLE MEMORY EMBEDDED IN A CENTRAL PROCESSING UNIT Examiner: Q. N. Nguyen

APPEAL BRIEF

MS Appeal Brief - Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Dear Sir:

As required under § 41.37(a), this brief is filed within two months of the Notice of Appeal filed in this case on June 21, 2005, and is in furtherance of said Notice of Appeal.

The fees required under § 41.20(b)(2) are dealt with in the accompanying TRANSMITTAL OF APPEAL BRIEF.

This brief contains items under the following headings as required by 37 C.F.R. § 41.37 and M.P.E.P. § 1206:

I. REAL PARTY IN INTEREST

The real party in interest for this appeal is BBNT Solutions LLC

II. RELATED APPEALS, INTERFERENCES, AND JUDICIAL PROCEEDINGS

There are no other appeals, interferences, or judicial proceedings which will directly affect or be directly affected by or have a bearing on the Board's decision in this appeal.

III. STATUS OF CLAIMS

Claims 1-16 and 18-21 are pending in this application.

Claims 1-16 and 18-21 were finally rejected in the Office Action, dated May 10, 2005, and are the subject of the present appeal. These claims are reproduced in the Claim Appendix of this Appeal Brief.

IV. STATUS OF AMENDMENTS

Applicant did not file an Amendment After Final Rejection.

V. SUMMARY OF CLAIMED SUBJECT MATTER

In the paragraphs that follow, each of the independent claims and means plus function claims that is involved in this appeal and each dependent claim that is argued separately will be recited followed in parenthesis by examples of where support can be found in the specification and drawings.

Claim 1 recites a central processing unit (CPU) (100, Fig. 1) in a network device. The central processing unit includes an arithmetic logic unit (140, Fig. 1); and a ternary content addressable memory operatively coupled to the arithmetic logic unit within the CPU and configured to perform one or more matching operations (260, Fig. 2; Fig. 5; pg. 10, para. 0030; pg. 15, para. 0042).

Claim 5 recites that the one or more matching operations include a packet stuff/unstuff operation (pg. 15, para. 0043; pg. 16, para. 0046).

Claim 7 recites that the ternary content addressable memory is located within the arithmetic logic unit (Fig. 2; pg. 7, para. 0023).

Claim 8 recites a first register configured to store a first 32-bit operand (310-380, Fig. 3; pg. 9, para. 0027; pg. 13, para. 0038); and a second register configured to store a second 32-bit operand (310-380, Fig. 3; pg. 9, para. 0027; pg. 13, para. 0038).

Claim 10 recites that the ternary content addressable memory includes a memory array including a group of 64-bit entries (pg. 11, para. 0032), and wherein, when performing the one or more matching operations, the ternary content addressable memory compares higher order bits of each entry of the memory array to the first 32-bit operand and compares lower order bits of each entry of the memory array to the second 32-bit operand (pg. 11, para. 0032).

Claim 16 recites a method for processing packets in a network device. The method includes receiving a packet (pg. 5, para. 0018; pg. 12, para. 0035; pg. 15, para. 0043); and processing the packet using a ternary content addressable memory resident within an arithmetic logic unit of the network device (pg. 15, paras. 0042 and 0043).

Claim 20 recites a system for forwarding packets in a network device. The system includes means for receiving at least one packet (130, Fig. 1; pg. 5, para. 18; pg. 6, para. 0021; pg. 12, para. 0035; pg. 15, para. 0043); and means for processing the packet using a ternary content addressable memory (260, Fig. 2) resident within a central processing unit (100, Fig. 1) of the network device (260, Fig. 2; 100, Fig. 1; 130, Fig. 1; pg. 6, para. 0021; pg. 15, paras. 0042 and 0043).

Claim 21 recites an arithmetic logic unit (140) comprising a register unit (250, Fig. 2); an operations unit (270, Fig. 2); and a ternary content addressable memory coupled to the register unit and the operations unit within the arithmetic logic unit (260, Fig. 2; pg. 9, para. 0028; pg. 10, para. 0030; pg. 11, para. 0034).

VI. GROUNDS OF OBJECTION TO BE REVIEWED ON APPEAL

A. Claims 1-16 and 18-21 stand rejected under 35 U.S.C. § 102(e) as anticipated by Nataraj et al. (U.S. Patent No. 6,757,779).

VII. ARGUMENT

A. The rejection of claims 1-16 and 18-21 under 35 U.S.C. § 102(e) as allegedly anticipated by Nataraj et al. (U.S. Patent No. 6,757,779) should be reversed.

The initial burden of establishing a *prima facie* basis to deny patentability to a claimed invention always rests upon the Examiner. In re Oetiker, 977 F.2d 1443, 24 USPQ2d 1443 (Fed. Cir. 1992). A proper rejection under 35 U.S.C. § 102 requires that a single reference teach every aspect of the claimed invention either explicitly or impliedly. Any feature not directly taught must be inherently present. Verdegaal Bros. v. Union Oil Co. of California, 814 F.2d 628, 2 USPQ2d 1051 (Fed. Cir. 1987).

1. Claims 1-4, 6, and 11-15.

With the above principles in mind, Appellant's claim 1 is directed to a central processing unit (CPU) in a network device. The CPU includes an arithmetic logic unit and a ternary content addressable memory operatively coupled to the arithmetic logic unit within the CPU and configured to perform one or more matching operations. Nataraj et al. does not disclose or suggest this combination of features.

For example, Nataraj et al. does not disclose or suggest an arithmetic logic unit. The Examiner relies on CAM blocks 1-K in Fig. 60 of Nataraj et al. as allegedly corresponding to an arithmetic logic unit (final Office Action, pp. 3 and 7). Appellant submits that these elements of Nataraj et al. do not correspond to an arithmetic logic unit.

Fig. 60 of Nataraj et al. corresponds to a partitionable CAM device with intra-row configurability embodiment of the Nataraj et al. invention (col. 58, line 51). In this embodiment,

Nataraj et al. discloses a CAM device that includes multiple independently-selectable CAM blocks 1-K, an instruction decoder 6019, address logic 6011, a global flag circuit 6031, a global priority encoder 6033, and a block control circuit 6027 (col. 58, lines 52-56). Each of CAM blocks 1-K includes a configurable CAM array 6001, a configurable priority index table 6003, configurable match logic and multiple match flag logic 6007, configurable priority encoder logic 6005, and mode-responsive read/write circuitry 6015. Nataraj et al. in no way discloses or suggests that one or more of CAM blocks 1-K are an arithmetic logic unit, as "arithmetic logic unit" is commonly known in the art. One skilled in the art would readily appreciate that an "arithmetic logic unit" is a specific kind of device (typically one with two inputs, a function select input, and one output containing the result of the function applied to the two inputs). A common example of an arithmetic logic unit is the old TTL 74181 arithmetic logic unit chip and its various descendants. One skilled in the art would readily appreciate that Nataraj et al.'s CAM blocks 1-K are not an arithmetic logic unit, as recited in claim 1. The Examiner has not pointed to any section of Nataraj et al. that discloses that any of CAM blocks 1-K is an arithmetic logic unit or explained why one skilled in the art would construe CAM blocks 1-K to be an arithmetic logic unit. In fact, Appellant notes that Nataraj et al. does not even disclose an arithmetic logic unit.

Further with respect to the above feature of claim 1, the Examiner alleges that "the language in the quotation 'an arithmetic logic unit' can be given a broadest and reasonable interpretation in light of the specification as a component of a microprocessor used for arithmetic, comparative, and logical functions (the definition of ALU from Microsoft Computer Dictionary - Fifth Edition" (final Office Action, pg. 7). Even with this definition in mind, the Examiner has not pointed to any section of Nataraj et al. that discloses that CAM blocks 1-K are a component of a microprocessor used for arithmetic, comparative, and logical functions. In

fact, Nataraj et al. specifically discloses that an external host processor issues instructions to CAM device 600 (see, for example, col. 58, lines 64-66). Therefore, given the fact that Nataraj et al. does not anywhere state that CAM device 6000 is implemented in a CPU, Nataraj et al.'s statement at col. 58, lines 64-66, provides explicit evidence that CAM device 600 is external to a CPU (i.e., the host processor) and, therefore, is not resident within a CPU. Therefore, even if one accepts the Examiner's definition for an arithmetic logic unit, Nataraj et al. appears to clearly teach that CAM device 600 is external to the processor (i.e., the CPU) and, thus, not a "component of a microprocessor," as the Examiner alleges. Also, the embodiment disclosed in connection with Nataraj et al.'s Fig. 75 clearly shows the processor as being a separate entity than the CAM device.

Nataraj et al. does not further disclose or suggest a ternary content addressable memory (CAM) operatively coupled to the arithmetic logic unit within the CPU and configured to perform one or more matching operations, as also recited in claim 1. The Examiner relies on element 6001 in Fig. 60, Fig. 4, col. 7, lines 38-65, col. 58, lines 52-63, and col. 60, line 62 to col. 61, line 8, of Nataraj et al. for allegedly disclosing the recited ternary CAM of Appellant's claim 1 (final Office Action, pp. 3 and 8). Appellant respectfully disagrees.

Element 6001 in Fig. 60 of Nataraj et al. corresponds to a CAM array. As set forth above, CAM array 6001 is part of a CAM block 1-K within CAM device 6000 that is disclosed according to a partitionable CAM device with intra-row configurability embodiment of the Nataraj et al. invention. Nataraj et al. in no way discloses or suggests, however, that CAM device 6000 (or CAM blocks 1-K) is a CPU. Therefore, Nataraj et al. cannot disclose or suggest a ternary CAM included within a CPU, as recited in claim 1.

Moreover, as set forth above, Nataraj et al. does not disclose an arithmetic logic unit. Therefore, Nataraj et al. cannot disclose CAM array 6001 being operatively coupled to an

arithmetic logic unit. Instead, Nataraj et al. specifically discloses CAM array 6001 being connected to mode-responsive read/write circuitry 6015, a priority index table 6003, comparand driver 6025, and address logic 6011 (see Fig. 60). Nataraj et al. in no way discloses or suggests that any of these devices is an arithmetic logic unit.

Fig. 4 of Nataraj et al. depicts one embodiment of a classification or filtering system 400 for a policy-based router (col. 7, lines 39-40). System 400 includes a CAM device 402 and a route memory 414. This figure of Nataraj et al. in no way discloses or suggests a ternary CAM operatively coupled to an arithmetic logic unit within a CPU and configured to perform one or more matching operations, as recited in claim 1.

At col. 7, lines 38-65, Nataraj et al. discloses:

> FIG. 4 shows one embodiment of a classification or filtering system 400 for a policy-based router. System 400 includes CAM device 402 and route memory 414. CAM device 402 includes policy statement table 404, priority index table 406 and encoder 412, and may be an integrated circuit component formed on a single semiconductor substrate.
>
> For this embodiment, policy statement table 404 is stored in a ternary CAM array that stores policy statements or policy words. Ternary CAM array 404 has rows of CAM cells 405 for storing policy field information PF1-PFX, where X is any number. Each policy field PF1-PFX can include any policy information including DA, SA, PTCL, TOS, and COST, or any other type of policy field to assist in the classification or filtering of the policy statement to provide a certain Quality of Service (QoS), Class of Service (CoS), and the like. Each policy field may include any number of bits. Additional information associated with each policy field may be stored in one or more additional binary or ternary CAM cells or other types of memory cells disposed in each row of CAM 404. Ternary CAM 404 also has rows of mask cells 407 for storing mask data M1-MX corresponding to each row of policy fields 405. Global masks (not shown) may be used to mask entire columns in CAM array 404 as generally known in the art. For alternative embodiments, CAM array 404 may be any other type of CAM including a binary CAM, or any other type of memory to store policy statements to be compared with processed policy information of an incoming packet.

This section of Nataraj et al. discloses a ternary CAM 404 that stores policy field information. This section of Nataraj et al. in no way discloses or suggests that CAM 404 is part of a CPU, as recited in claim 1, or is operatively coupled to an arithmetic logic unit, as also recited in claim 1.

At col. 58, lines 52-63, Nataraj et al. discloses:

> FIG. 60 illustrates an embodiment of a CAM device 6000 that includes multiple independently selectable CAM blocks, 1-K, instruction decoder 6019, address logic 6011, global flag circuit 6031, global priority encoder 6033, and a block control circuit 6027. As shown by the exploded view of CAM block 1, each of the CAM blocks includes a configurable CAM array 6001, configurable priority index table 6003, configurable match flag logic and multiple match flag logic (depicted in FIG. 60 as a block flag circuit 6007), configurable priority encoder logic (depicted in FIG. 60 as a block priority encoder 6005), and mode-responsive read/write circuitry 6015, all as described above in reference to FIGS. 4-59.

This section of Nataraj et al. discloses that CAM blocks 1-K include a CAM array 6001. This section of Nataraj et al. in no way discloses or suggests that CAM array 6001, which the Examiner alleges corresponds to the recited ternary CAM, is included within a CPU or is operatively coupled to an arithmetic logic unit, as recited in claim 1.

At col. 60, line 62 to col. 61, line 8, Nataraj et al. discloses:

> Referring again to FIG. 60, in one embodiment, each block configuration signal 6018_1 -6018_K is a multi-bit signal that indicates one of a number of word-width configurations for the CAM array of the corresponding CAM block (e.g., each block configuration signal includes the configuration signals SZ32, SZ64, SZ128 and SZ256 discussed above). As with the configuration signal, CFG, in the single CAM block architecture described above in reference to FIG. 15, each block configuration signal 6018 is supplied to the read/write circuit 6015, priority index table 6003, flag logic 6007 and priority encoder 6005 within the corresponding CAM block to support block-level generation of read/write control signals, qualified match signals, match flag, multiple match flag, full flag and match address signals for the different CAM array configurations.

This section of Nataraj et al. discloses a block configuration signal 6018 that is supplied to each CAM block 1-K. This section of Nataraj et al. in no way discloses or suggests that CAM array

6001, which the Examiner alleges corresponds to the recited ternary CAM, is included within a CPU or is operatively coupled to an arithmetic logic unit, as recited in claim 1.

Nataraj et al. does not disclose all of the features of claim 1. Therefore, Nataraj et al. cannot anticipate claim 1.

For at least the foregoing reasons, Appellant submits that the rejection of claim 1 under 35 U.S.C. § 102(e) based on Nataraj et al. is improper. Accordingly, Appellant requests that the rejection be reversed.

Since claims 2-4, 6 and 11-15 depend from claim 1, Appellant requests that the rejection of those claims be reversed for at least the reasons given above with respect to claim 1.

2. Claim 5.

Claim 5 depends from claim 1. Therefore, claim 5 is not anticipated by Nataraj et al. for at least the reasons given above with respect to claim 1. Moreover, this claim recites an additional feature not disclosed or suggested by Nataraj et al.

Claim 5 recites that the one or more matching operations include a packet stuff/unstuff operation. The Examiner relies on col. 14, lines 27-63, of Nataraj et al. for allegedly disclosing the above feature of claim 5 (final Office Action, pg. 3). Appellant respectfully disagrees.

At col. 14, lines 27-63, Nataraj et al. discloses:

> Inserting and Deleting Policy Statements
>
> As previously described, priority numbers for policy statements may be assigned in ascending or descending priority order such that there are gaps left between the numbers to accommodate the new priority numbers associated with new policy statements to be stored in CAM array 404. Alternatively, the priority numbers may be assigned in consecutive ascending or descending priority order. New policy statements and their associated priority numbers can be added to the tables 404 and 408 in conformance with either assignment method without having to reload or physically reorder the policy statements or the priority numbers in the respective tables.
>
> Each new policy statement can be loaded into any location (e.g., the next free

> location) in CAM array 404, and can be assigned a new priority number without having to reload or reorder CAM array 404 and priority memory 408. When a new policy statement is received, its priority number can be compared with the existing priority numbers already stored in priority memory 408 to determine if a policy statement already exists that has been assigned that priority. It is generally desirable that no two policy statements have the same priority number. Thus, if the priority number already exists, the network administrator or the policy-based router itself can assign the new policy statement a new priority number, or the priority number of the existing policy statement can be updated (i.e., incremented or decremented). Since the existing priority numbers are stored in ascending or descending order, updating one priority number may also result in the need to update other priority numbers such that no two priority numbers are the same.
>
> When a policy statement is deleted from CAM array 404, the corresponding priority number in priority memory 408 is also deleted. The policy statements and priority numbers can be deleted by setting one or more valid bits to an appropriate state for the row of CAM array 404 that stores the policy statement to be deleted. The valid bit(s) may be stored in CAM array 404, priority memory 408, or in each of the memory arrays.

This section of Nataraj et al. discloses the ability to insert and delete policy statements into CAM array 404. This section of Nataraj et al. in no way relates to one or more matching operations that includes a packet stuff/unstuff operation, as recited in claim 5.

For at least the foregoing reasons, Appellant submits that the rejection of claim 5 under 35 U.S.C. § 102(e) based on Nataraj et al. is improper. Accordingly, Appellant requests that the rejection be reversed.

3. Claim 7.

Claim 7 depends from claim 1. Therefore, claim 7 is not anticipated by Nataraj et al. for at least the reasons given above with respect to claim 1. Moreover, this claim recites an additional feature not disclosed or suggested by Nataraj et al.

Claim 7 recites that the ternary content addressable memory is located within the arithmetic logic unit. The Examiner points to CAM array 6001 as allegedly corresponding to the recited ternary content addressable memory and CAM block 1 as allegedly implemented as an arithmetic logic unit (final Office Action, pg. 4). Appellant disagrees.

Nataraj et al.'s Fig. 60 depicts a CAM device 6000 that includes multiple independently selectable CAM blocks 1-K. Contrary to the Examiner's allegation, Nataraj et al. in no way discloses or suggests that any of CAM blocks 1-K is implemented as an arithmetic logic unit. Therefore, the disclosure of Nataraj et al. does not support the Examiner's allegation. Moreover, the Examiner does not explain why one skilled in the art would reasonably construe Nataraj et al.'s CAM block 1 as an arithmetic logic unit. Accordingly, a *prima facie* case of anticipation has not been established with respect to claim 7.

As set forth above with respect to claim 1, one skilled in the art would readily appreciate that an "arithmetic logic unit" is a specific kind of device (typically one with two inputs, a function select input, and one output containing the result of the function applied to the two inputs). A common example of an arithmetic logic unit is the old TTL 74181 arithmetic logic unit chip and its various descendants. Nataraj et al. in no way discloses or suggests that any of CAM blocks 1-K is implemented as an arithmetic logic unit and the Examiner has not pointed to any section of Nataraj et al. that supports this allegation.

For at least the foregoing reasons, Appellant submits that the rejection of claim 7 under 35 U.S.C. § 102(e) based on Nataraj et al. is improper. Accordingly, Appellant requests that the rejection be reversed.

4. Claims 8 and 9.

Claim 8 depends from claim 1. Therefore, claim 8 is not anticipated by Nataraj et al. for at least the reasons given above with respect to claim 1. Moreover, this claim recites an additional feature not disclosed or suggested by Nataraj et al.

Claim 8 recites a first register configured to store a first 32-bit operand and a second register configured to store a second 32-bit operand. The Examiner relies on elements C1-C8 in Nataraj et al.'s Fig. 21 as allegedly corresponding to the recited first and second registers and

points to col. 37, lines 46-62, of Nataraj et al. for support (final Office Action, pg. 4). Appellant disagrees.

Elements C1-C8 in Nataraj et al.'s Fig. 21 correspond to comparand register segments (col. 37, lines 50-52). Nataraj et al. discloses that comparand register segments C1-C8 store comparand segments (col. 37, lines 50-52). Contrary to the Examiner's allegation, Nataraj et al. in no way discloses or suggests that a first comparand register segment of comparand register segments C1-C8 stores a first 32-bit operand and a second comparand register segment of comparand register segments C1-C8 stores a second 32-bit operand, as recited in claim 8.

At col. 37, lines 46-62, Nataraj et al. discloses:

> FIG. 21 illustrates an embodiment of a comparand load circuit which may be used in the exemplary CAM device described above in reference to FIGS. 16-19 (i.e., 64-bit wide data bus 1604, and a CAM array 1601 that includes Z=8 row segments (S1-S8) per row, each row segment having W=32 CAM cells). A comparand register 2103 includes eight comparand register segments, C1-C8, to store as many as eight corresponding comparand segments. Comparand register segments C1, C3, C5 and C7 are coupled to receive comparand data from the lower 32 signal lines of the data bus 1604 (i.e., signal path 2106), while comparand register segments C2, C4, C6 and C8 are coupled to receive comparand data from a multiplexer circuit 2105 via signal path 2108. When the CAM array 1601 is in a x32 configuration, the multiplexer circuit 2105 selects the lower 32 signal lines of the data bus to provide comparand data to comparand register segments C2, C4, C6 and C8, such that all eight comparand register segments are coupled to receive the same 32-bit value from the data bus 1604.

This section of Nataraj et al. discloses that a comparand register 2103 includes eight comparand register segments C1-C8. As set forth above, Nataraj et al. does not disclose or suggest that a first comparand register segment of comparand register segments C1-C8 stores a first 32-bit operand and a second comparand register segment of comparand register segments C1-C8 stores a second 32-bit operand, as recited in claim 8.

For at least the foregoing reasons, Appellant submits that the rejection of claim 8 under 35 U.S.C. § 102(e) based on Nataraj et al. is improper. Accordingly, Appellant requests that the rejection be reversed.

Since claim 9 depends from claim 8, Appellant requests that the rejection of claim 9 be reversed for at least the reasons given above with respect to claim 8.

5. Claim 10.

Claim 10 depends from claim 8. Therefore, claim 10 is not anticipated by Nataraj et al. for at least the reasons given above with respect to claim 8. Moreover, this claim recites an additional feature not disclosed or suggested by Nataraj et al.

Claim 10 recites that the ternary content addressable memory includes a memory array including a group of 64-bit entries. Claim 10 also recites that, when performing the one or more matching operations, the ternary content addressable memory compares higher order bits of each entry of the memory array to the first 32-bit operand and compares lower order bits of each entry of the memory array to the second 32-bit operand. The Examiner relies on col. 37, line 15 to col. 38, line 24, of Nataraj et al. for allegedly disclosing these features (final Office Action, pg. 4). Appellant disagrees.

At col. 37, line 15 to col. 38, line 24, Nataraj et al. discloses:

> In the ZY/2x2W mode (i.e., two row segments per group), the CSSEL signals cause select logic 2001 to enable the odd CEN signals CEN1, CEN3, etc. such that the same first portion of comparand data is written into the first comparand segments associated with the first row segments S1, S3, etc. of CAM array 1501. In a subsequent cycle, the CSSEL signals cause select logic 2001 to enable the even CEN signals CEN2, CEN4, etc. such that the same second portion of comparand data is written into the second comparand segments associated with the second row segments S2, S4, etc. The first and second portions of comparand data together form the entire (2W) comparand data. This methodology continues until, in the YxZW mode, the CEN signals are sequentially enabled to consecutively load each portion (W) of the ZW comparand data into one of the Z comparand segments. The operation of this embodiment is further illustrated by the example of FIG. 21.

> FIG. 21 illustrates an embodiment of a comparand load circuit which may be used in the exemplary CAM device described above in reference to FIGS. 16-19 (i.e., 64-bit wide data bus 1604, and a CAM array 1601 that includes Z=8 row segments (S1-S8) per row, each row segment having W=32 CAM cells). A comparand register 2103 includes eight comparand register segments, C1-C8, to store as many as eight corresponding comparand segments. Comparand register segments C1, C3, C5 and C7 are coupled to receive comparand data from the lower 32 signal lines of the data bus 1604 (i.e., signal path 2106), while comparand register segments C2, C4, C6 and C8 are coupled to receive comparand data from a multiplexer circuit 2105 via signal path 2108. When the CAM array 1601 is in a x32 configuration, the multiplexer circuit 2105 selects the lower 32 signal lines of the data bus to provide comparand data to comparand register segments C2, C4, C6 and C8, such that all eight comparand register segments are coupled to receive the same 32-bit value from the data bus 1604. When the CAM array 1601 is configured for x64, x128 or x256 operation, the multiplexer circuit 2105 selects the upper 32 signal lines of the data bus to provide comparand data to comparand register segments C2, C4, C6 and C8, such that comparand register segment pairs C1|C2, C3|C4, C5|C6 and C7|C8 are coupled to receive a 64-bit data value from the data bus 1604. In the embodiment of FIG. 21, the multiplexer circuit is controlled by the configuration signal, SZ32 (a component of the CFG signal) to select either the lower or upper half of the data bus 1604 to source data for the even numbered comparand register segments.
>
> Comparand enable signals, CEN[8:1], are generated in accordance with the configuration signals (i.e., SZ32, SZ64, SZ128 and SZ256) and comparand segment select signals CSSEL1 and CSSEL0 to enable selected comparand register segments to be loaded with comparand data. More specifically, the configuration signals indicate the size of an incoming comparand word (i.e., x32, x64, x128 or x256) and, when the incoming comparand word is larger than the data bus (i.e., a x128 long comparand word or x256 long comparand word), the CSSEL1 and CSSEL0 signals are used to load a 64-bit component of the long comparand word into the appropriate pair of comparand register segments. In one embodiment, when the comparand word is a 64-bit value (i.e., SZ=64), the 64-bit comparand word is loaded into all four comparand register segment pairs simultaneously. Similarly, when the comparand word is a 32-bit value (i.e., SZ=32), the 32-bit comparand word is loaded into all eight comparand register segments simultaneously.

This section of Nataraj et al. discloses the loading of comparand data into comparand register segments C1-C8. This section of Nataraj et al. in no way discloses or suggests that, when performing one or more matching operations, the ternary content addressable memory compares higher order bits of each entry of the memory array to the first 32-bit operand and compares

lower order bits of each entry of the memory array to the second 32-bit operand, as recited in claim 10.

For at least the foregoing reasons, Appellant submits that the rejection of claim 10 under 35 U.S.C. § 102(e) based on Nataraj et al. is improper. Accordingly, Appellant requests that the rejection be reversed.

6. Claims 16, 18, and 19.

Claim 16 is directed to a method for processing packets in a network device. The method includes receiving a packet; and processing the packet using a ternary content addressable memory resident within an arithmetic logic unit of the network device. Nataraj et al. does not disclose or suggest this combination of features.

For example, Nataraj et al. does not disclose or suggest processing a packet using a ternary content addressable memory resident within an arithmetic logic unit of a network device. With respect to claim 16, the Examiner alleges "[c]laims 16 and 18-19 are corresponding method claims of CPU claims 1, 3 and 6; therefore, they are rejected under the same rationale" (final Office Action, pg. 6). Appellant submits that claims 1, 3, and 6 do not recite, however, processing a packet using a ternary content addressable memory resident within an arithmetic logic unit of a network device, as recited in claim 16. Accordingly, a *prima facie* case of anticipation has not been established with respect to claim 16.

Nonetheless, Nataraj et al. does not disclose or suggest processing a packet using a ternary content addressable memory resident within an arithmetic logic unit of a network device, as recited in claim 16. With respect to claim 1, the Examiner relies on CAM blocks 1-K as allegedly corresponding to an arithmetic logic unit and on CAM array 6001 in Nataraj et al.'s Fig. 60 as allegedly corresponding to a ternary content addressable memory (final Office Action, pg. 3). Appellant respectfully disagrees with this interpretation of Nataraj et al.

Fig. 60 of Nataraj et al. corresponds to a partitionable CAM device with intra-row configurability embodiment of the Nataraj et al. invention (col. 58, line 51). In this embodiment, Nataraj et al. discloses a CAM device that includes multiple independently-selectable CAM blocks 1-K, an instruction decoder 6019, address logic 6011, a global flag circuit 6031, a global priority encoder 6033, and a block control circuit 6027 (col. 58, lines 52-56). Each of CAM blocks 1-K includes a configurable CAM array 6001, a configurable priority index table 6003, configurable match logic and multiple match flag logic 6007, configurable priority encoder logic 6005, and mode-responsive read/write circuitry 6015. Nataraj et al. in no way discloses or suggests that one or more of CAM blocks 1-K are an arithmetic logic unit, as "arithmetic logic unit" is commonly known in the art. One skilled in the art would readily appreciate that an "arithmetic logic unit" is a specific kind of device (typically one with two inputs, a function select input, and one output containing the result of the function applied to the two inputs). A common example of an arithmetic logic unit is the old TTL 74181 arithmetic logic unit chip and its various descendants. One skilled in the art would readily appreciate that Nataraj et al.'s CAM blocks 1-K are not an arithmetic logic unit, as recited in claim 16. Nataraj et al. in no way discloses or suggests that any of CAM blocks 1-K is implemented as an arithmetic logic unit and the Examiner has not pointed to any section of Nataraj et al. that supports this allegation or explained why one skilled in the art would construe Nataraj et al.'s CAM blocks 1-K as an arithmetic logic unit.

Since Nataraj et al. does not disclose or suggest an arithmetic logic unit, Nataraj et al. cannot disclose or suggest a ternary content addressable memory resident within the arithmetic logic unit, as also recited in claim 16. Therefore, Nataraj et al. cannot disclose or suggest processing a packet using a ternary content addressable memory resident within an arithmetic logic unit of a network device, as recited in claim 16.

Nataraj et al. does not disclose all of the features of claim 16. Therefore, Nataraj et al. cannot anticipate claim 16.

For at least the foregoing reasons, Appellant submits that the rejection of claim 16 under 35 U.S.C. § 102(e) based on Nataraj et al. is improper. Accordingly, Appellant requests that the rejection be reversed.

Claims 18 and 19 depend from claim 16. Therefore, Appellant requests that the rejection of these claims be reversed for at least the reasons given above with respect to claim 16.

7. Claim 20.

Claim 20 is directed to a system for forwarding packets in a network device. The system includes means for receiving at least one packet and means for processing the packet using a ternary content addressable memory resident within a central processing unit of the network device. Nataraj et al. does not disclose or suggest this combination of features.

For example, Nataraj et al. does not disclose or suggest means for processing the packet using a ternary content addressable memory resident within a central processing unit of the network device. The Examiner does not specifically address the features of claim 20. Instead, the Examiner references the rejection of claim 16 (final Office Action, pg. 6). Claim 16, however, does not recite means for processing a packet using a ternary content addressable memory resident within a central processing unit of a network device, as recited in claim 20. Instead, claim 16 recites processing a packet using a ternary content addressable memory resident within an arithmetic logic unit of a network device. The Examiner does not address the specific features recited in claim 20. Accordingly, a *prima facie* case of anticipation has not been established with respect to claim 20.

Nonetheless, Nataraj et al. does not disclose a central processing unit. The Examiner does not point to any section of Nataraj et al. that corresponds to a central processing unit.

Therefore, the Examiner has not established a *prima facie* case of anticipation with respect to claim 20.

The Examiner alleges, with respect to claim 1, that Nataraj et al.'s CAM array 6001 (Fig. 60) corresponds to a ternary content addressable memory. With that interpretation in mind, Nataraj et al. does not disclose or suggest that CAM array 6001 is resident within a central processing unit, as recited in claim 20. Instead, Nataraj et al. discloses that CAM array 6001 is located within a CAM block 1 in a CAM device 6000. Nataraj et al. in no way discloses or suggests that CAM block 1 or CAM device 6000 is a central processing unit. Therefore, Nataraj et al. cannot disclose or suggest means for processing a packet using a ternary content addressable memory resident within a central processing unit of a network device, as recited in claim 20.

Nataraj et al. does not disclose all of the features of claim 20. Therefore, Nataraj et al. cannot anticipate claim 20.

For at least the foregoing reasons, Appellant submits that the rejection of claim 20 under 35 U.S.C. § 102(e) based on Nataraj et al. is improper. Accordingly, Appellant requests that the rejection be reversed.

8. Claim 21.

Claim 21 is directed to an arithmetic logic unit. The arithmetic logic unit includes a register unit; an operations unit; and a ternary content addressable memory coupled to the register unit and the operations unit within the arithmetic logic unit. Nataraj et al. does not disclose or suggest this combination of features.

For example, Nataraj et al. does not disclose or suggest an arithmetic logic unit. The Examiner relies on CAM block 1-K of CAM device 6000 in Nataraj et al.'s Fig. 60 as allegedly corresponding to an arithmetic logic unit (final Office Action, pg. 6).

Fig. 60 of Nataraj et al. corresponds to a partitionable CAM device with intra-row configurability embodiment of the Nataraj et al. invention (col. 58, line 51). In this embodiment, Nataraj et al. discloses a CAM device 6000 that includes multiple independently-selectable CAM blocks 1-K, an instruction decoder 6019, address logic 6011, a global flag circuit 6031, a global priority encoder 6033, and a block control circuit 6027 (col. 58, lines 52-56). Each of CAM blocks 1-K includes a configurable CAM array 6001, a configurable priority index table 6003, configurable match logic and multiple match flag logic 6007, configurable priority encoder logic 6005, and mode-responsive read/write circuitry 6015. Nataraj et al. in no way discloses or suggests that CAM blocks 1-K are an arithmetic logic unit, as "arithmetic logic unit" is commonly known in the art. One skilled in the art would readily appreciate that an "arithmetic logic unit" is a specific kind of device (typically one with two inputs, a function select input, and one output containing the result of the function applied to the two inputs). A common example of an arithmetic logic unit is the old TTL 74181 arithmetic logic unit chip and its various descendants. One skilled in the art would readily appreciate that Nataraj et al.'s CAM blocks 1-K are not an arithmetic logic unit, as recited in claim 21. The Examiner has not pointed to any section of Nataraj et al. that discloses that any of CAM blocks 1-K is an arithmetic logic unit or explained why one skilled in the art would construe Nataraj et al.'s CAM blocks 1-K as an arithmetic logic unit. Appellant notes that Nataraj et al. does not even disclose an arithmetic logic unit.

Since Nataraj et al. does not disclose or suggest an arithmetic logic unit, Nataraj et al. cannot disclose an arithmetic logic unit that includes a ternary content addressable memory coupled to a register unit and an operations unit within the arithmetic logic unit, as also recited in claim 21.

Nataraj et al. does not disclose all of the features of claim 21. Therefore, Nataraj et al. cannot anticipate claim 21.

For at least the foregoing reasons, Appellant submits that the rejection of claim 21 under 35 U.S.C. § 102(e) based on Nataraj et al. is improper. Accordingly, Appellant requests that the rejection be reversed.

VIII. CONCLUSION

In view of the foregoing arguments, Appellant respectfully solicits the Honorable Board to reverse the Examiner's rejection of claims 1-16 and 18-21 under 35 U.S.C. § 102.

Applicant believes no fee is due with this response other than as reflected on the enclosed Appeal Brief Transmittal. However, if a fee is due, please charge our Deposit Account No. 18-1945, under Order No. BBNT-P01-128 from which the undersigned is authorized to draw.

Dated: August 2, 2005

Respectfully submitted,

By _____

Edward A. Gordon
Registration No.: 54,130
ROPES & GRAY LLP
One International Place
Boston, Massachusetts 02110-2624
(617) 951-7000
(617) 951-7050 (Fax)
Attorneys/Agents For Applicant

APPENDIX A

Claims Involved in the Appeal of Application Serial No. 09/938921

1. In a network device, a central processing unit (CPU) comprising:

an arithmetic logic unit; and

a ternary content addressable memory operatively coupled to the arithmetic logic unit within the CPU and configured to perform one or more matching operations.

2. The CPU of claim 1 wherein the one or more matching operations includes a network packet processing operation.

3. The CPU of claim 2 wherein the packet processing operation includes an address lookup operation.

4. The CPU of claim 3 wherein the address lookup operation includes an Internet Protocol (IP) address lookup operation.

5. The CPU of claim 1 wherein the one or more matching operations includes a packet stuff/unstuff operation.

6. The CPU of claim 1 wherein the one or more matching operations includes a packet classification operation.

7. The CPU of claim 1 wherein the ternary content addressable memory is located within the arithmetic logic unit.

8. The CPU of claim 1 further comprising:

a first register configured to store a first 32-bit operand; and

a second register configured to store a second 32-bit operand.

9. The CPU of claim 8 wherein the ternary content addressable memory performs the one or more matching operations based on at least one of the first or second 32-bit operands.

10. The CPU of claim 8 wherein the ternary content addressable memory includes a memory array including a group of 64-bit entries, and

wherein, when performing the one or more matching operations, the ternary content addressable memory compares higher order bits of each entry of the memory array to the first 32-bit operand and compares lower order bits of each entry of the memory array to the second 32-bit operand.

11. The CPU of claim 1 wherein the ternary content addressable memory includes a memory array that includes a group of 64-bit entries.

12. The CPU of claim 11 wherein the memory array comprises 32 entries.

13. The CPU of claim 1 wherein, when performing the one or more matching operations, the ternary content addressable memory is configured to:

compare an operand to a group of entries.

14. The CPU of claim 13 wherein the ternary content addressable memory is further configured to:

set a first flag when the operand fails to match an entry in the group of entries, and

set a second flag when the operand matches multiple entries of the group of entries.

15. The CPU of claim 13 wherein, prior to comparing, the ternary content addressable memory is configured to:

sequentially load the group of entries from a succession of mask/value pairs transferred to the ternary content addressable memory.

16. A method for processing packets in a network device, comprising:

receiving a packet; and

processing the packet using a ternary content addressable memory resident within an arithmetic logic unit of the network device.

18. The method of claim 16 wherein the processing includes performing a matching operation using information in a header of the packet.

19. The method of claim 18 wherein the processing includes a packet classification operation.

20. A system for forwarding packets in a network device, comprising:

means for receiving at least one packet; and

means for processing the packet using a ternary content addressable memory resident within a central processing unit of the network device.

21. An arithmetic logic unit comprising:

a register unit;

an operations unit; and

a ternary content addressable memory coupled to the register unit and the operations unit within the arithmetic logic unit.